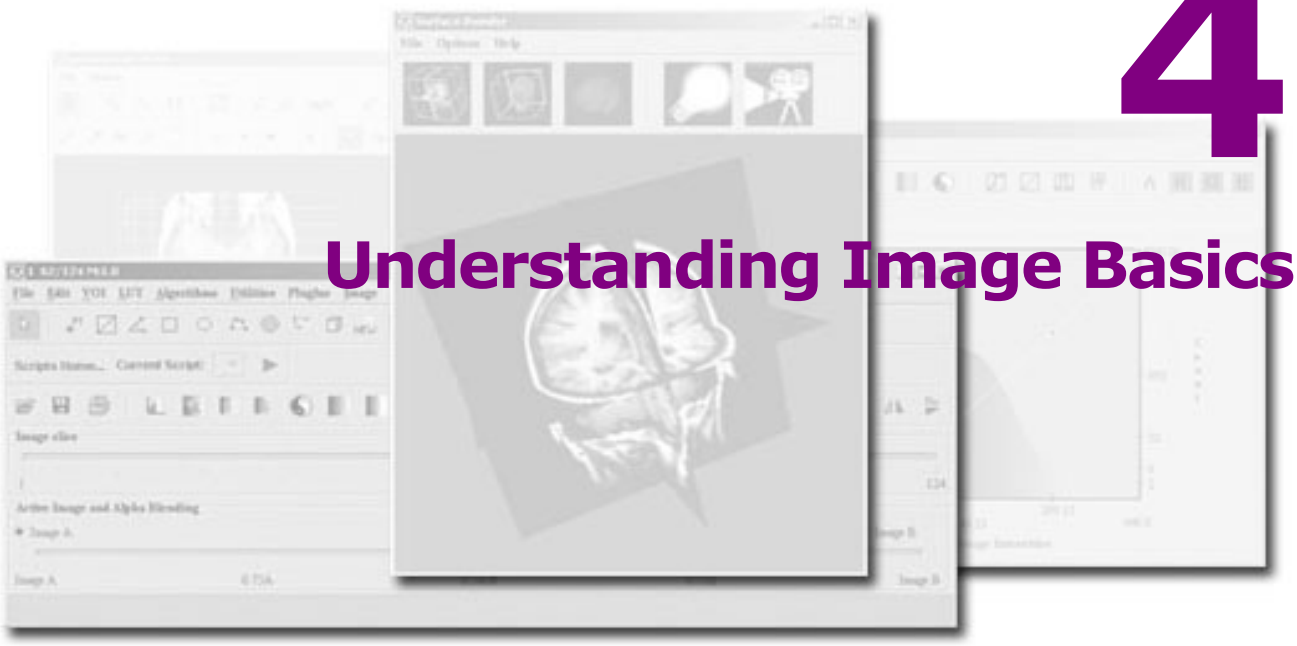
4



Understanding Image Basics

In this chapter . . .

This chapter provides basic information about image file formats. It presents an overview of graphic file formats and the formats that MIPAV supports. It also explains the dimensionality and coordinate system used by MIPAV, the data types MIPAV supports, the header offset information, endianness, and how to view and edit image attributes.

Supported graphic file formats

MIPAV supports three categories of graphic file formats: vector (i.e., volume of interest, or VOI), bitmap (i.e., 2D to 5D grayscale or RGB images), and 3D images.

Vector file formats

Vector file formats are usually used to store line-based elements. These elements can be geometric shapes, such as polygons, curves, or splines. The data portion of the vector file contains a mathematical description of elements of an image with specific key points identified. A key point can be thought of as a *node* or *vertex;* it marks the place where the line changes direction. For example, a square may contain four key points while a circle may contain many more.

In addition to key points, the vector file usually indicates the line segment's starting point, direction, and length. In MIPAV you can create geometrically shaped contours to indicate VOIs. You can then store the VOIs in a vector formatted file. MIPAV can read and generate several vector-formatted file types including the MIPAV lookup table (LUT), MTX, Graphics Plot (PLT), and VOI.

Bitmap file formats

Bitmap file formats are usually used to store real-world graphics data, such as photographs or medical images such as CT scans. The data portion of bitmap files contains numerical data that indicates the color of each pixel or voxel in the image. The bitmap file may also indicate the *bit depth,* which indicates the number of colors that a pixel or voxel can represent. Typically, the bit depth can be 1, 2, 4, 8, 15, 16, 24, or 32. The number of colors represented is the square of the bit depth. Thus, a 4-bit image can contain 16 colors; an 8-bit image can contain 64 colors; and so on. MIPAV can read and generate a number of bitmap formatted file types, such as Adobe Photoshop (PSD), Graphics Interchange File (GIF), Sun Raster (RS), Tagged Image File Format (TIFF), and Truevision Graphics Adapter (TGA).

Three-dimensional (3D) graphic file formats

MIPAV uses 3D graphic formatted files to store descriptions of the color and shape of 3D models of real-world and imaginary objects.

Supported file formats

To support the wide range of image-processing needs of the NIH intramural research community, MIPAV reads and writes the image file types indicated in Table 2 on page 111. This table shows the formal name of the file format, the file extensions associated with the format, the numerical format, and whether MIPAV supports the encoding or decoding of the format.

Table 2. Graphics file formats supported by MIPAV

File format and extensions	E	D	Supports	Does not support
Adobe Photoshop (PSD)	Y	Y	Encoding of bitmap, grayscale, RGB, and index mode images. Also supports the encoding of RLE compression for all image variants. Decoding of Adobe Photoshop v 2.5 images (and above) with the following modes: bitmap, grayscale, palette, and RGB.	Alpha saving
Analyze	Y	Y	TBD	TBD
Audio Video Interleave (AVI)	Y	Y	Uncompressed RLE8	TBD
Cheschire (IMG or IMC)	N	Y	TBD	TBD
DICOM	Y	Y	TBD	TBD
Graphics Interchange File (GIF)	N	Y	Decoding of interlaced images and images with a transparency information set.	Encoding of a GIF file Decoding of multiframe GIFs using the multiframe load mechanism
Joint Photographics Experts Group (JPEG, JFIF, JFI, JPG)	Y	Y	TBD	TBD
Medical Image NetCDF (MINC)	Y	Y	TBD	TBD
MIPAV LUT	Y	Y	TBD	TBD
MIPAV MTX	Y	Y	TBD	TBD
MIPAV PLT	Y	Y	TBD	TBD
MIPAV VOI	Y	Y	TBD	TBD

Table 2. Graphics file formats supported by MIPAV (continued)

File format and extensions	E	D	Supports	Does not support
Microsoft Windows Bitmap (BMP, DIB)	Y	Y	Encoding of IndexColorModel images as uncompressed BMP palette formats. Other ColorModel images are encoded as uncompressed 24-bit BMP formatted files. Decoding of BMP version 2.x through 4.x files. Also supports the encoding of OS/2 variant files. Supports the encoding of true color and palette images, as well as Run-Length Encoding 8 (RLE8) and RLE4 images.	Decoding of old versions of BMP files (prior to version 2.x)
PC Paintbrush File Format (PCX, DCX, PCC)	Y	Y	Encoding of PCX files including uncompressed and RLE-compressed support. Full support for decoding of PCX files.	N/A
PICT	Y	Y	Encoding of black and white images as bitmap mode PICTs. Encodes grayscale images from 2 to 8-bits as palette mode (PICT format does not have grayscale mode). Encodes palette images as palette mode PICTS, encodes other images as a 24-bit 3 component directed bits PICT. Decoding of black and white and up to (and including) 32-bit color PICT images. Also supports the decoding of compressed PICT images.	N/A
Portable Network Graphic Format (PNG)	Y	Y	Encodes index images as palette PNG images. Also encodes grayscale formatted images (which include grayscale and direct color) with bit masks saved in grayscale file format. Direct color and RGB are encoded as RGB images with an alpha mask (if mask is available from the image). 16-bit depth images are truncated to 8-bit depth when the image is loaded. Decoding of palette, grayscale, and true color images. Also supports the transparency chunk for palette based images.	Some extension chunks (such as ALPHA)
Quick Time (QT)	Y	Y	TBD	TBD
RAW	Y	Y	TBD	TBD
Sun Raster (RS, RAS)	Y	Y	Encodes 8-bit index color images with or without RLE compression or RGB format for other color models. Decoding of 4, 8, 16, 24, and 32-bit images. Supports the decoding of the old, standard, byte encoded, RGB, TIFF, and IFF format type.	Decoding of RAW colormap or experimental type images
Tag Image File Format (TIFF)	Y	Y	TBD	Tiled

Table 2. Graphics file formats supported by MIPAV (continued)

File format and extensions	E	D	Supports	Does not support
Truevision Graphics Adapter (TGA, VST, VDA, ICB, TPIC)	Y	Y	Encodes index color images in an uncompressed TGA palette format; encodes other images in an uncompressed 24-bit TGA format. Decodes black and white, palette, true color uncompressed images. Decodes 16-bit, 24-bit, and 32-bit true color variants (can include loading of alpha channel).	N/A
X BitMap (XBM)	Y	Y	Encoding of 2-color XBM images. Fully supports the decoding of XBM formatted files.	N/A
X PixMap (XPM)	Y	Y	Encoding of XPM palette images. Fully supports the decoding of XPM formatted files.	Decoding of image files with 3 or more characters per pixel/ voxel

Adobe Photoshop (PSD) files

PSD is a bitmap file format created by Adobe. It can accommodate images with an unlimited amount of colors. PSD files are not multiple image files; the maximum image size is 30K x 30K pixels or voxels. PSD files are generally used to store images that were altered or manipulated by the Adobe Photoshop software. The header is embedded in the file and is 26 bytes in length. It contains information such as the height and width of the image and the color mode.

Analyze files

Analyze formatted files are generated by a UNIX-based, image-processing application developed at the Mayo Clinic. Analyze file format supports:

- 1 bit (packed binary)
- 8 bit (8 bits per pixel or voxel [unsigned byte])
- 16 bit (16 bits per pixel or voxel [signed short])
- 32 bit (32 bits per pixel or voxel [signed integers, or floating point])
- 64 bit (64 bits per pixel or voxel [doubles, or complex])
- 24 bit (RGB, 8 bits per channel red, green, blue)

Analyze images are formed from two files:

- *Header file*—This file describes the image type, size, and other important image attributes.
- *File Containing the Actual Image Data*—This file can be interpreted as a raw file since it does not have any header information within the file.

These files have the same name and are distinguished by the extensions *.hdr* for the header file and *.img* for image file. For example, *brain.hdr* is the header file that describes the image file and *brain.img* is the file that contains the image data. In support of some NIH intramural researchers who have used Analyze in the past and have generated a large number of legacy datasets, MIPAV reads and writes Analyze formatted images.

Audio Video Interleave (AVI) files

AVI is the Microsoft Video for Windows standard. AVI is a form of the Resource Interchange File Format (RIFF). In this file format, video and audio data are stored consecutively in an AVI file. The AVI file contains a 4-byte file header, followed by list information and then alternating video and audio streams.

Digital Imaging and Communications in Medicine (DICOM) files Medicine (DICOM) files

The DICOM format is a standard that defines a standard method of communication between two devices, such as computers, servers, or imaging devices. DICOM not only prescribes how to communicate with other imaging equipment and databases but also specifies how images are stored. Each image has accompanying header information describing the image format (i.e., height, width, etc.) as well as information that indirectly relates to the image (i.e., patient information, image equipment setup parameters).

Graphics Interchange Format (GIF) files

The Graphics Interchange Format (GIF) is a bitmap file format that was created by CompuServe. GIF is primarily an exchange and storage file format; GIFs can be used to store one or several bitmap images in one file. GIFs support pixel or voxel depths of 1 to 8 bits. The maximum image size is

64K x 64K pixel or voxels. There are two major revisions of the GIF format specification: GIF87a and GIF89a. Both formats begin with a 6-byte header that identifies the file format as GIF.

Joint Photographic Experts Group (JPEG/JFIF) files

The JPEG File Interchange Format (JFIF) allows files containing JPEG-encoded data streams to be exchanged. Technically, JPEG refers to the compression type and the Joint Photographic Experts Group standards organization. However, the term *JPEG* is usually used to indicate the file format. JPEG file formats are bitmap files that are primarily used in image and graphics manipulation programs. Created by C-Cube Microsystems, the JPEG or JFIF file format does not accommodate multiple images per file and the maximum image size allowed is 64K by 64K pixel or voxels. JPEG files can accommodate 24-bit color images. Generally, JPEG header information appears between the Start of Image (SOI) and Application (APP0) markers.

Microsoft Windows Bitmap (BMP) files

BMP is a native bitmap format for the Microsoft Windows and OS/2 platforms. Developed by the Microsoft Corporation, it can be read on Intel machines running Microsoft Windows, Windows XP, Windows NT, Windows 95, OS/2, and MS-DOS. It is an uncompressed file format with a maximum image size of 32K x 32K or 2G x 2G pixels or voxels (depending on the version of BMP).

Depending on the version of BMP, for OS/2 the maximum image size is larger, 64K x 64K or 4G x 4G pixels or voxels. There are several versions of BMP for Microsoft Windows and OS/2. MIPAV supports Microsoft Window BMP versions 2.x and above and all OS/2 versions of the BMP file format. BMP version 2.x is designed for use with the Microsoft Windows 2.x platform. It has a 14-byte header (as does the OS/2 1.x bitmap header). It can accommodate images with 1-, 4-, 8-, or 24-bit colors. Version 3.x is designed for use with the Microsoft Windows 3.x and Windows NT platforms. Like version 2.x, it contains a 14-byte header. It also contains an additional bitmap header that is 40 bytes in size. The Microsoft Windows 3.x platform version accommodates images with 1-, 4-, 8- or 24-bit colors. Windows NT and Windows XP accommodate 16- and 32-bit images. Version 4.x was designed for use with Microsoft Windows 95. It contains

the 14-byte header, and an additional 108-byte bitmap header. It can accommodate images of 1, 4, 8, 16, and 32 bits.

Medical Image NetCDF (MINC) files

MINC is a medical-imaging file format that is based on the Network Common Data Form (NetCDF) file format. NetCDF is a platform-independent software interface that provides a means for storing named, multidimensional variables. Each multidimensional variable is defined by a name, dimensions, and attributes. For example, cardio (name) [256, 256] (dimensions), and "long_name" where this attribute is a string that describes the content of the image. MINC provides a standard for dimension, variable, and attribute names suitable for medical imaging. MINC also provides convenience functions to complement the NetCDF interface and convenience functions for using MINC files.

MIPAV (LUT) files

LUT is a vector file format that is used to store lookup table data. A sample file appears in Appendix C.

MIPAV (MTX) files

A sample MTX file appears in Appendix C.

MIPAV (PLT) files

PLT is a vector file format that is used to store graphics data. A sample file appears in Appendix C.

MIPAV (VOI) files

VOI is a vector file format that is used to store volume of interest contouring information. A sample file appears in Appendix C.

PC Paintbrush (PCX) files

PC Paintbrush is a bitmap file format that is used primarily in Microsoft Windows and other Windows-based products. PCX is mainly used as an

exchange and storage format. Created by ZSoft and packaged with Microsoft Windows, PCX can accommodate 1-, 4-, 8-, and 24-bit color images. It is uncompressed; the maximum image size is 64K x 64K pixel/voxels. Header information is embedded in the file; the first 128 pixel/voxels in the file contain information such as the PCX ID number, the bits per pixel/voxel, and the palette type.

PICT: TBD files

Portable Network Graphic Format (PNG) files

PNG is a bitmap file format that is generally used to transmit and store network image data. The PNG format can store images with up to 16-bits (grayscale) or 48-bits (truecolor) per pixel/voxel. The maximum image size is 2G x 2G pixel/voxels. PNG is a compressed format. Multiple images cannot be stored in PNG. An 8-byte identification signature is followed by a header chunk, which contains basic information about the image data. The header chunk data area is 13 bytes in length.

Raw data files

MIPAV supports reading and writing Raw image data of all the basic programming types (i.e., boolean, byte, short, etc.). Raw images have no header or a header of known length and unknown content at a fixed location at the beginning of the file. You must specify basic information about the Raw image before it is loaded in MIPAV. When loading a RAW data set, you must specify the image type, dimension and resolution, units, and header offset.

Sun Raster (RS) files

RS files is the native bitmap format for Sun Microsystems UNIX platforms that run the SunOS operating system. This format stores bitmap data (color, grayscale, black and white) of any pixel/voxel depth. RS files can be of any size; however, multiple images per file are not supported. The header is embedded in the file; it is 32 bytes in the length and contains typical header information such as the width and height of the image and the type of color map used.

Tag Image File Format (TIFF) files

TIFF is a bitmap file format that is generally used to provide a portable, image-storage mechanism that describes image data. Multiple images can be stored in one file. TIFF files can accommodate a maximum image size of $2^{32} - 1$ pixel/voxels. It is widely used and is a standard file format used in many desktop publishing, imaging, and paint programs. MIPAV supports most TIFF 6.0 formatted files used commonly in the image research community including 2D and 3D monochrome 8-bit, signed and unsigned 16-bit, and signed 32-bit images. Future support of 24-bit color and compressed images is planned. The TIFF Image File Header (IFH) is 8 bytes in length and contains 3 fields of information. If multiple images are in a file, an IFH is present for each image in the file.

Truevision Graphics Adapter (TGA) files

TGA is a bitmap file format that is often used in graphics, imaging, and paint applications that store up to 32-bit color images (8-, 16-, 24-, and 32-bit colors are supported). Created by Truevision, Incorporated, it is an uncompressed file format that does not support multiple images per file. The maximum image size however, is unlimited. TGA's header is 18 bytes in length and contains traditional header information including the depth of the color map entries.

X BitMap (XBM) files

XBM is a monochrome bitmap file format that is used primarily for the storage of icon and cursor bitmaps in X Windows. It was developed by X Consortium. Because this format was developed for small amounts of data, the bitmap images are composed of collections of ASCII data rather than binary data. (XBM bitmap data is often found in C source header files.) Multiple images can be stored on a file; there is no limit to the image size. XBM has no header file, nor is a formal header embedded in the program; rather header information consists of four lines that begin with `#define`. The four lines of code indicate the height and width of the image and the coordinate of the hotspot (if any).

X PixMap (XPM) files

XPM is a bitmap file format developed by Groupe Bull. It is used primarily to store X Windows pixmap information. XPM can store black and white, grayscale or color image data of unlimited colors. Like XBM, there is no limit for the image size and there can be multiple images per file. In addition, like XBM, XPM is written in ASCII. It can contain a section, `<Values>`, that contains header information, such as the height and width of the pixmap and the number of characters per pixel or voxel.

Understanding the MIPAV dimensionality and coordinate system

Internally, MIPAV is designed to store images of any dimensionality and thus is *n* dimensional. Figure 81 is a model of an example 3D-image where each position within the grid represents a *voxel*.

voxel—The smallest distinguishable cube-shaped part of a 3D-image.

The voxel in the upper left-most position is considered the origin of the dataset. (A volumetric dataset is a set of cross-sectional images). Ideally, each voxel would be isotropic (i.e., cubic), but clinical datasets are often acquired in such a manner that the resolution in the x and y dimensions is greater than the resolution in the z dimension, thus producing rectangular voxels (elongated in the z direction). This voxel anisotropy is taken into account in the execution of the algorithms found in MIPAV.

Because MIPAV's image storage buffer is *n*-dimensional, it can store 2D, 3D, and 4D+ medical images. CT and MRI are examples of 3D medical images where the three dimensions (x, y, and z) are spatial dimensions. Other 3D datasets include fluoroscopy volumes, where the first two dimensions are spatial and the third dimension is time.

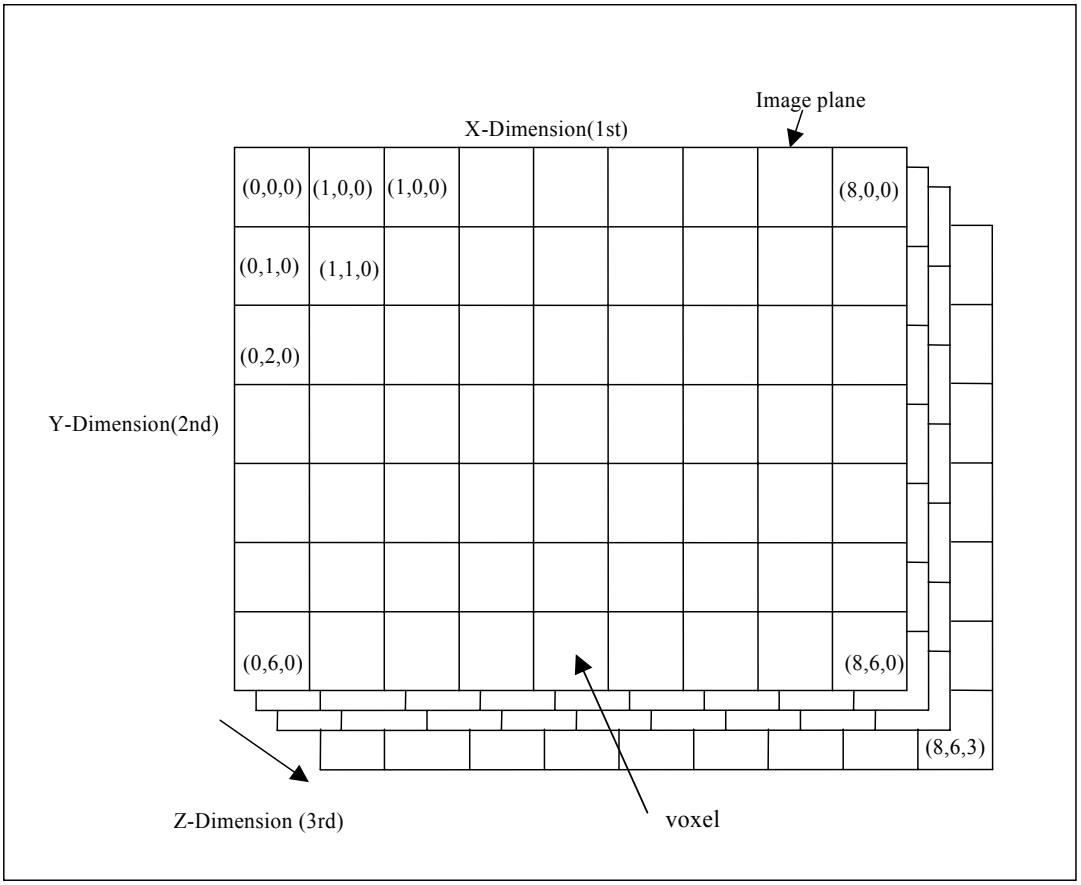


Figure 81. Example of 3D image (origin at top left)

Data types

A *data type* is the classification of a particular type of information. For example, a floating point data type indicates a number with a decimal point. Data read from image files are stored in MIPAV's data structures. MIPAV supports all the basic data types shown in Table 3and may in the future support more composite data types including color and complex numbers. The image data read from the file is stored in a data buffer of the same basic data type, preventing the reduction of image information that results from conversion into a fixed data type.

Table 3. Data types supported by MIPAV

Data type	Description
Boolean	1 bit per pixel/voxel (1 on, 0 off)
Signed byte	1 byte per pixel/voxel (-128, 127)
Unsigned byte	1 byte per pixel/voxel (0, 255)
Signed short	2 bytes per pixel/voxel (-32768, 32767)
Unsigned short	2 bytes per pixel/voxel (0, 65535)
Integer	4 bytes per pixel/voxel (-2^{31}, $2^{31}-1$)
Long	8 bytes per pixel/voxel (-9.22E18, 9.22E18)
Float	4 bytes per pixel/voxel (-3.4E38, 3.4E38)
Double	8 bytes per pixel/voxel (-1.8E308, 1.8E308)
Color 32	3 bytes per pixel/voxel, plus 1 byte; 8 bits per color channel (alpha, red, green, and blue)

Headers

If you attempt to open a raw image, you may need to supply MIPAV with the *header offset*. The header offset indicates the size of the space reserved at the beginning of the file where specific types of information is kept. This space, which is called the header, precedes the image data. If you know the length of the header, enter it in the header offset text box. When MIPAV accesses the file, it skips the header offset and begins to read the image data.

Note: Not all image file formats have a header.

Endianness

Endianness refers to the byte ordering of the data. Some computers order the data with the least significant byte (LSB) first followed by the most significant byte (MSB). This byte order is referred as "little endian" or Intel byte ordering. machines that use little-endian byte ordering are VAXes, Intel x86, and Pentium. The reverse is MSB then LSB, which is referred as "big endian" or Motorola byte ordering. Machines that use big-endian byte

ordering are IBM System 3D, RISC, and a Motorola 680x0. MIPAV is biendian; that is, it supports both big- and little-endian byte-ordering formats.

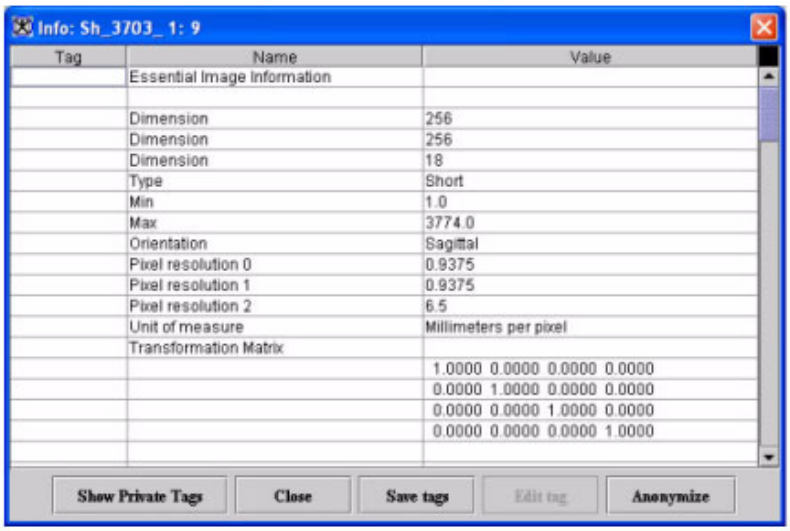
Viewing and editing image attributes

Information about the image appears in the Image Information dialog box and in the Image Attributes dialog box.

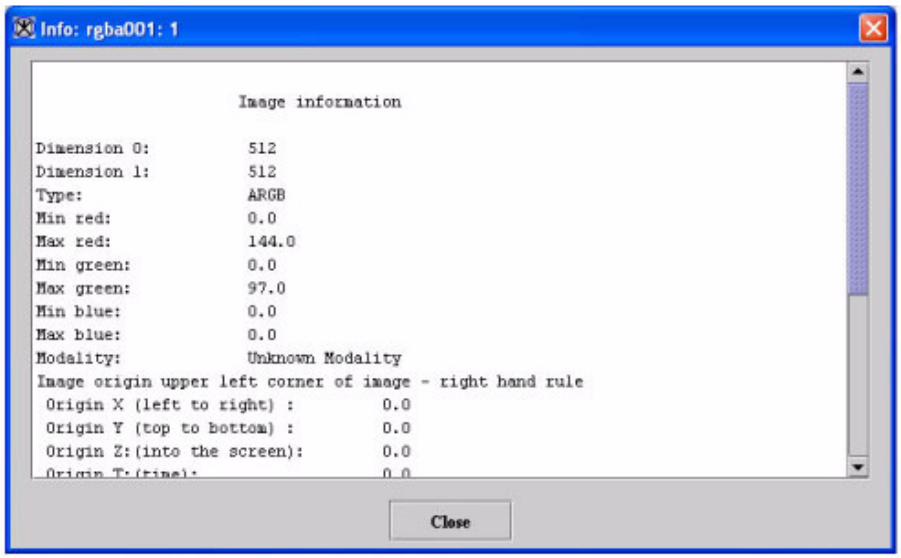
The Image Attributes window displays the name of the image file and information about the resolution of the image. The About Image window displays more detailed information about the image, such as the orientation of the image or the number of bits per pixel or voxel.

Viewing image attributes

To view the attributes of an image, select Image > Attributes > View Header in the MIPAV window. For DICOM datasets, the Image Information dialog box similar to the one shown in Figure 82a appears. For non-DICOM datasets, an Image Information dialog box similar to the one in Figure 82b appears.



(A) Information dialog box for a DICOM image



(B) Information dialog box for a nonDICOM image

Figure 82. Info dialog boxes for both DICOM and nonDICOM images

Editing image attributes

You can edit some of the image attributes for both DICOM datasets and non-DICOM datasets in the Image Attributes dialog box (Figure 83). The

Image Attributes dialog box includes five tabbed pages, which are the following:

- **Name page**—Displays the name of the image (without its extension, or suffix) and lists the image modality
- **Resolutions page**—Indicates the unit of measure for each dimension
- **Orientations page**—Lists the image orientations
- **Dataset Origin**—Lists the origin for the first image slice
- **Transform Matrix**—Displays and lists the values for the transform matrix

By default, the Name page always appears first when the Image Attributes opens.

To modify the name of the image dataset

1. Open either a DICOM or nonDICOM image dataset.
2. Select Image > Attributes > Edit Attributes in the MIPAV window. The Image Attributes dialog box opens.

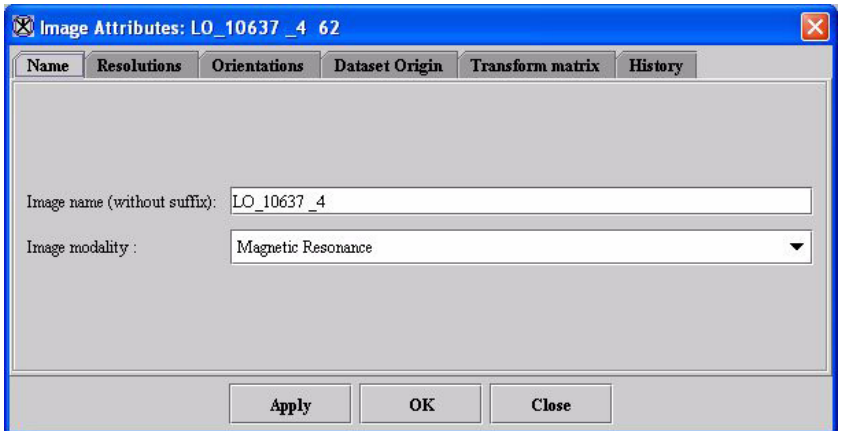


Figure 83. The Name page in the Image Attributes window

3. Select the name of the image in the Image name (without suffix) box.
4. Type a new name of the image.
5. Click either Apply or OK. The new image name appears in the header on the image window.

What's the Difference Between Clicking Apply or Clicking OK?

Click Apply when you want the dialog box or window to remain open after it makes the change so that you can make further modifications. If instead you click OK, the dialog box or window closes immediately after making the current change.

To edit the dimensions and units of measure

1. Open either a DICOM or nonDICOM image dataset.
2. Select Image > Attributes > Edit Attributes in the MIPAV window. The Image Attributes dialog box opens.
3. Click Resolutions. The Resolutions page in the Image Attributes dialog box appears.
4. Type the changed values in the dimension text boxes on the left and then select the appropriate value in the Unit of measure boxes on the right.
5. Click either Apply or OK.

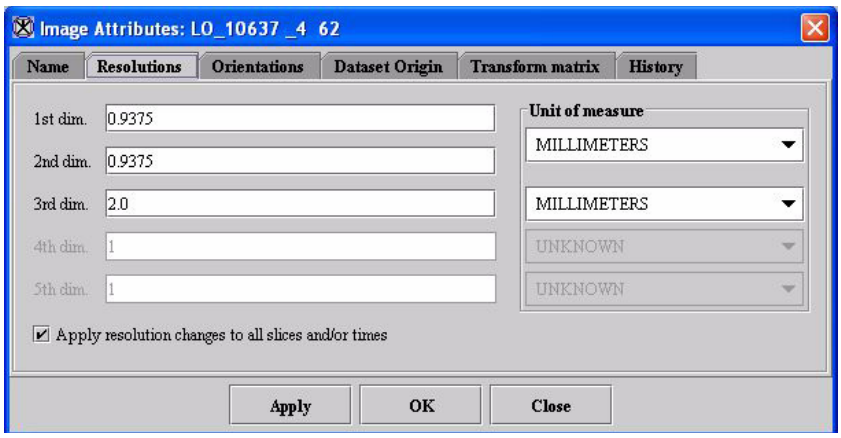


Figure 84. The Resolutions page in the Image Attributes dialog box

To edit the *x*, *y*, and *z* orientations of image datasets

1 Open either a DICOM or nonDICOM image dataset.

2 Select Image > Attributes > Edit Attributes in the MIPAV window. The Image Attributes dialog box opens.

3 Click Orientations. The Orientations page (Figure 85) in the Image Attributes dialog box opens.

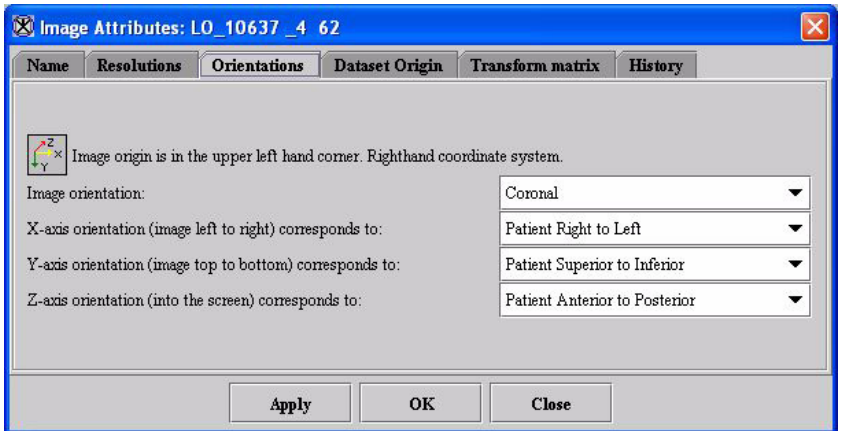


Figure 85. Orientations page in the Image Attributes dialog box

4 Select the appropriate values in the boxes on the right.

5 Click Apply or OK.

To change the origin for the first image slice

1 Open either a DICOM or nonDICOM image dataset.

2 Select Image > Attributes > Edit Attributes in the MIPAV window. The Image Attributes dialog box opens.

3 Click Dataset Origin. The Dataset Origin page (Figure 86) in the Image Attributes dialog box appears.

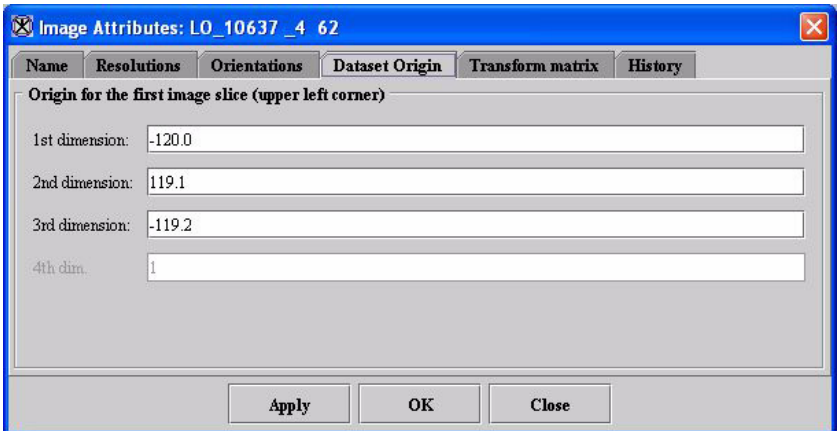


Figure 86. Dataset Origin page in the Image Attributes dialog box

4 Type the appropriate values in the dimension boxes.

5 Click Apply or OK.

To change the transform matrix

1 Open either a DICOM or nonDICOM image dataset.

2 Select Image > Attributes > Edit Attributes in the MIPAV window. The Image Attributes dialog box opens.

3 Click Transform Matrix. The Transform Matrix page (Figure 87) in the Image Attributes dialog box appears.

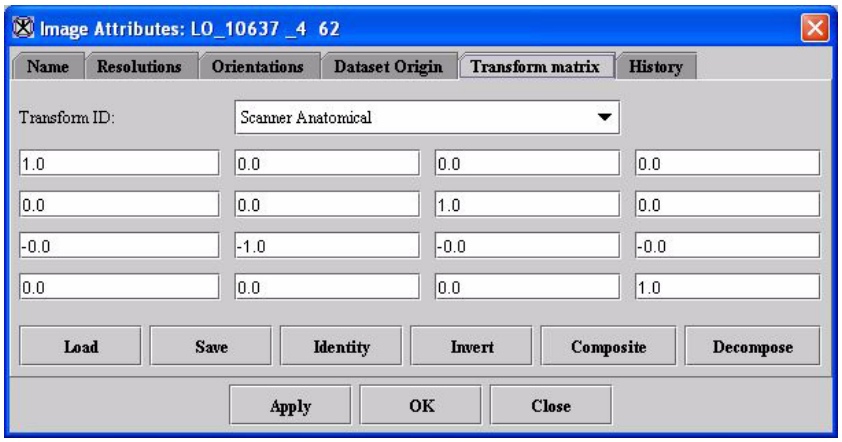


Figure 87. Transform Matrix page in the Image Attributes dialog box

4 Select the appropriate transform ID in the Transform ID box.

5 Load and save transform matrices.

6 Click Apply or OK.